Assignment 6: Dynamic Parallelism

- Extend a raytracer by scattered reflections
- A reflection causes $16 * 16$ new rays for each parent ray
  - The result is added (with weighting) to the parent ray result color
- DP kernel dimensions
  - blockDim = $(16,16)$
  - gridDim = #parentRay = N (only count rays that contribute notably)

Assignment 6: Dynamic Parallelism

- Fast counting algorithm, that can also generate indices?
  - Could be computed in shared memory

```
extern __shared__ T *sharedData;
__global__ void kernel( ... ) {
        ...
}
int memsize = ...; // determine how much data is needed
kernel<<<x,y,memsize>>>( ... );
```

- Multiple arrays in dynamic shared memory:

```
int *buffer1 = &sharedData[0];
int *buffer2 = &sharedData[someNumber];
```

- memsize needs to be twice `someNumber`

Assignment 6: Dynamic Parallelism

- Data transfer from parent- to child-kernel and back
  - Transfer parent ray data (N Rays)
  - Transfer child kernel result (N Vec3s)
  - Block id can be used by scattered ray thread to read from and write to
  - => all threads per block share same parent ray & reflection result sum

```
parentKernel:
        __shared__ T *data; // share pointer among block
        data = (T*) malloc(N * sizeof(T)); // allocate global memory (once per block)
        childKernel<<<x,y>>>( data, ... ); // use memory to transfer data to or from
        // ...
        free(data); // (once per block)

childKernel:
        auto y = data[x]; // read from global memory
        data[x] = y; // write to global memory
```

Assignment 6: Dynamic Parallelism

- Scatter function:
  - $x_i = s\frac{x(t_i)}{x(d)-1} - \frac{s}{2}$, $y_i = s\frac{y(t_i)}{y(d)-1} - \frac{s}{2}$, $x_i^2 + y_i^2 + z_i^2 = 1$
  - t_i: threadId i
  - d: blockDim
  - $s \in [0,1]$
- Construct new orthogonal basis around ray.dir $\vec{r}$
  - $\vec{x} = \vec{r} \times \vec{v}, \quad \vec{r} \neq \vec{v}$
  - $\vec{y} = \vec{x} \times \vec{r}$
  - $\vec{z} = \vec{r}$
- Above basis not normal
  - Normalize vectors constructed with it
  - $\vec{r}_i = \frac{\vec{x}x_i + \vec{y}y_i + \vec{z}z_i}{|\vec{x}x_i + \vec{y}y_i + \vec{z}z_i|}$